Machine Learning techniques for state recognition and auto-tuning of quantum dot devices

[Updated: 2022-02-24]

Current semiconductor-based quantum computing approaches use adjacent, isolated electron spins as quantum bits. Establishing a stable configuration of electrons in space is achieved via electrostatic confinement, band-gap engineering, and dynamically adjusted voltages on nearby electrical gates. A key task is to determine a good set of control parameters (gate voltages) necessary to achieve the desired charge configuration – in both electron number and location – for a successful experiment. See Figure 1 for a generic model of a semiconductor heterostructure quantum dot (QD) device.

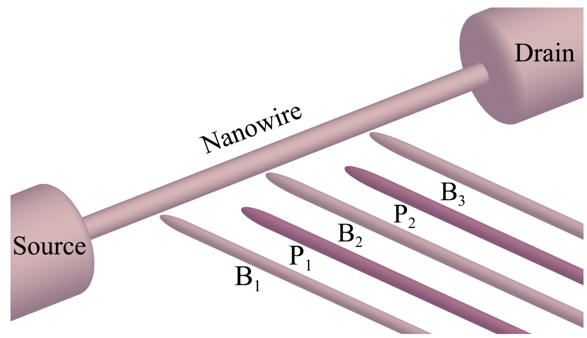

Figure 1: A generic model of a nanowire with five gates. Dots are formed in the nanowire due to modifications of the potential by adjustment of the barrier (B_i) and plunger (P_i) gates.

In recent years, machine learning (ML) has emerged as a "go-to" technique for automated data characterization, giving a reliable output when trained on a representative, comprehensive data set. In our work, we focus on the problem of classification of data into categories (also referred to as a *classification problem*). The categories can be either predefined and stored as *labels* (supervised learning) or extracted from the data (unsupervised learning). The ML algorithm learns about the categories from the training set and produces a model that can assign *previously unseen* inputs to those categories. Artificial neural networks (ANNs) – an ML approach inspired by analogous biological units – turned out to be particularly suitable for this task. ANNs were originally intended to mimic the workings of a human's brain. They are composed of (so-called) *artificial neurons* arranged in a series of fully connected layers (thus the term "deep neural networks"), with each layer performing a specific transformation of the data. Deep neural networks (DNNs), with multiple hidden layers, can be used to classify complex data into categories with high accuracy. One architecture that has performed particularly well in these settings are the convolutional neural networks (CNNs), that is networks where a set of convolutional and pooling layers precede the series of hidden layers. Each convolutional layer consists of a number of sets of weights (determined by the training on the dataset) that get convolved with the input. To learn larger scale features in the input more efficiently, a convolutional layer is generally followed by a pooling layer that takes in a sub-region in the input and replaces it by an effective element (e.g., the maximum element) in that region.

In our project, we use tools from ML to develop good heuristics for the gate voltages from a training set and design an effective and efficient neural network for the characterization of experimental data in semiconductor QD experiments. In the first stage of our project, we used a series of idealized (i.e., noiseless) simulated measurements to establish training and evaluation data sets. We use a modified Thomas-Fermi approximation to model a reference semiconductor system comprising a quasi-1D nanowire with a series of depletion gates whose voltages determine the number of islands and the charges on each of those islands, as well as the conductance through the 1D channel, see Figure 1.

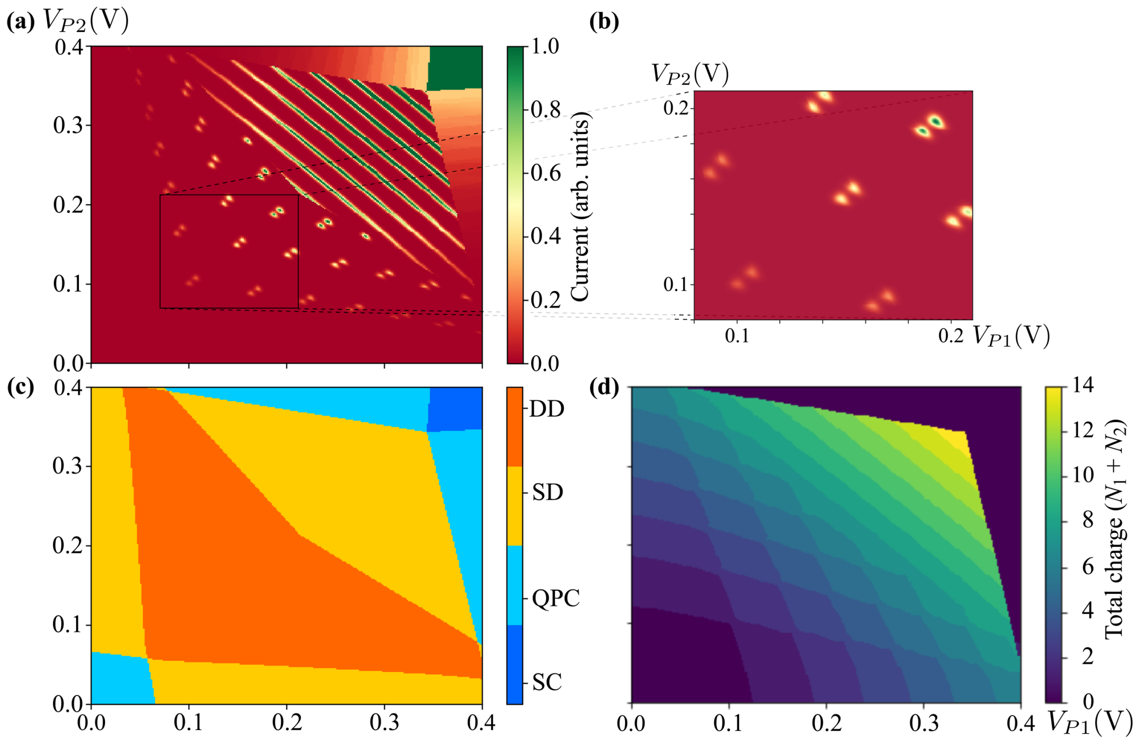

*Figure 2. An example of full 2D map (100x100 points) from the space of plunger gate voltages (V_{P1}, V_{P2}) to current **(a)** and a sub-region with double dot configuration **(b)**. A state map **(c)** and charge map **(d)** corresponding to the current map presented in (a). N_1 and N_2 denote the number of charges on each dot.*

For QFlow 1.0 dataset, we generated 1000 idealized simulated measurement with gate configurations averaging over different possible fabrications of the same type of device, with plunger voltages ranging from 0 to 400 mV. Each sample data is stored as a 100 x 100-pixel map from plunger voltages to current, net charge, sensor, and state maps (each stored separately). These maps can be used as observables and labels for training and testing. An example of a simulated device with current (a, b), state (c), and charge (d) is shown in Figure 2. The training and evaluation set was generated by taking 25 random 30 x 30-pixel sub-images of each map, resulting in 25 000 effective measurement realizations. We then trained a CNN model to "recognize" the electronic state within QD arrays (based both on the current and on the charge sensor data). We found > 95 % agreement between the CNN characterization and the Thomas-Fermi model predictions. Having the network trained, we tested it on the experimental data and found that it correctly identifies single and double QD state configurations, as well as the intermediate cases. We also showed a proof-of-concept implementation of a ML-driven auto-tuner in the multidimensional gate voltage space. We showed that by recasting the tuning problem as an optimization problem over the nanowire state in the space of gate voltages we can obtain a desired configuration of the QD system.

With QFlow 2.0 dataset, we aimed to extend the ML tools to nonideal devices by incorporating into the dataset various types of realistic, physical noise. To do this, we first generated 1 599 noiseless measurement using various gate configurations averaging over different possible fabrications of the same type of quasi-1D device, with plunger voltages ranging from -150 to 450 mV. This larger gate range allowed us to capture device states corresponding

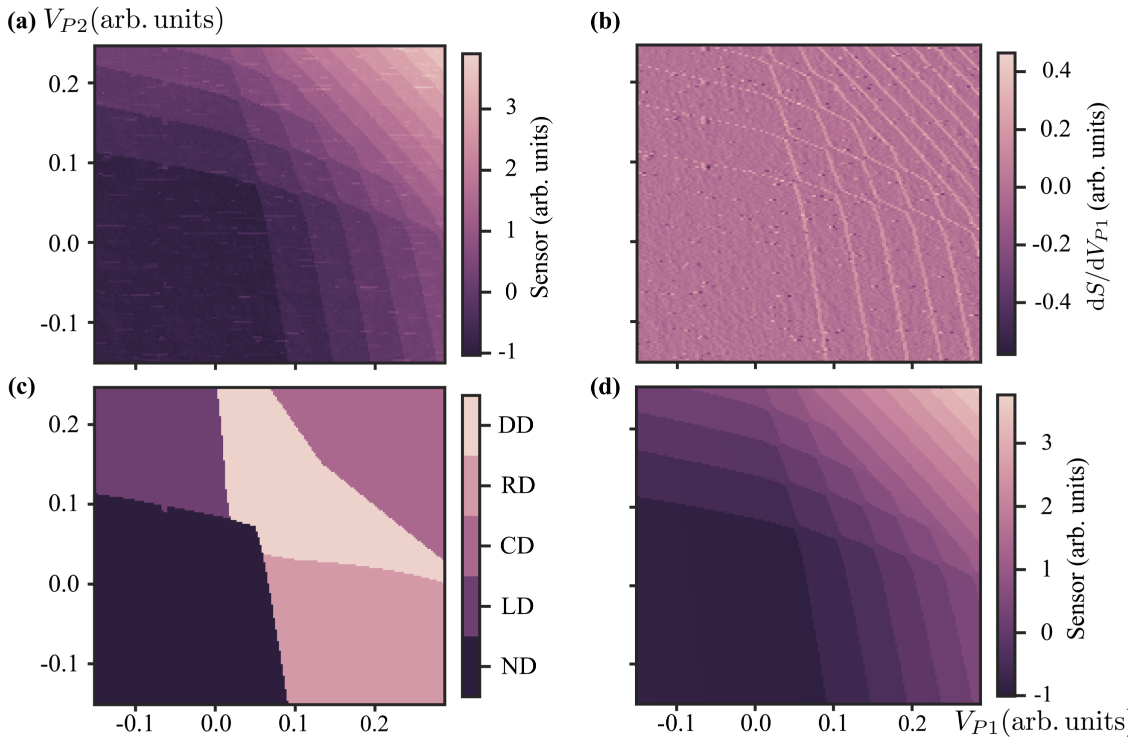

Figure 3. An example simulated measurement in the space two plunger gates. ***(a)*** *Sensor output with an optimized level of noise added.* ***(b)*** *Gradient of the sensor data in (a) in the x direction.* ***(c)*** *State of the device between no dot (ND), left dot (LD), central dot (CD), right dot (RD), and double dot (DD)* ***(d)*** *Noiseless version of the sensor data presented in (a).*

to single QD nearby each plunger gate in addition to the central single QD captured in QFlow 1.0 dataset. Each sample data is stored as a roughly 250 x 250-pixel map from plunger voltages to net charge, sensor, and state maps (each stored separately). Like in the QFlow 1.0 dataset, these maps can be used as observables and labels for training and testing. To match the qualitative features of real sensor maps more closely, we developed noise models to augment the data with realistic noise features. In particular, we added white noise, 1/f noise, dot jumps and sensor jumps. White and 1/f noise come from different kinds of electronic noise in the device. Dot and sensor jumps model fluctuations in the device state due to two level systems capturing or releasing electrons. The parameters defining the simulated noise were tuned using a semi-structured grid search. An example charge stability diagram from a simulated device with noise added is shown in Figure 3.

QFlow 2.0 dataset includes two distinct sets of data with the simulated noises as well as a manually labeled dataset of small experimental measurements and several unlabeled large-range experimental scans. The first simulated dataset has a range of signal-to-noise (SNR) encountered in moderately noisy devices. The actual noise level used to generate this dataset was set by randomly sampling the noise magnitudes over a Gaussian distribution with mean at 1.5 times the optimized values and standard deviation one third of the values (half of the optimized values). The second simulated dataset has a uniform distribution of noise magnitudes from 0 to 7 times the optimized magnitude to approximate the range of SNR values that could be encountered in real devices. In both dataset the relative magnitudes of white noise, 1/f noise, and sensor jumps are fixed and varied together between each simulated charge stability diagram to emulate a changing SNR. We add a different level of noise to each simulated device 10 times, taking the dataset from 1 599 devices to 15 990. The training and evaluation sets were generated from the noisy simulated datasets by taking 10 random 30 x 30-pixel sub-images of each map to go from 15 990 to 159 900 effective simulated measurement realizations.

Having generated data, we trained CNNs to either recognize the electronic state of QD devices based on the noisy sensor data or to recognize the level of noise in an image. We found roughly 94 % agreement between the CNN state characterization and the Thomas-Fermi model predictions for the electronic state. We found roughly 84 % agreement between the CNN noise characterization and the noise class determined from the amount of noise added.

We also tested both types of CNN on the experimental dataset. The small measurements were acquired from two devices fabricated on Si/Si_xGe_{1-x} heterostructures in an accumulation mode overlapping aluminum gate architecture. From one device we obtain two datasets of 82 and 503 images. From the other device we obtain 171 images. In total, the dataset consists of 756 hand-labeled experimental images. We tested our trained models on the experimental data and found that the CNN for state classification correctly identifies single and double dot state configurations, as well as the intermediate cases. Moreover, the CNN for noise characterization predicts lower signal-to-noise for images that the state classifier has more trouble identifying.